

CDF Future Plans for Databases and Application

Dmitry Litvintsev
(CD/CDF, Fermilab)

November 20, 2003

CDF DB “taking stock” meeting



Manpower

- **CDF/CDF ~ 1.5 FTE**
Dmitry Litvintsev - SPL
Randy Herber
- **CDF ~ 1.5 FTE**
Petar Maksimovic - Deputy SPL
Barry Blumenfeld, Andrew Hamilton, Larry Kirsch, Matt Martin, Mark Mathis, Tom Wright
- **CEPA/DBS ~ 4 FTE**
Lee Lueking (Head)
Dennis Box, Yuyi Guo, Margherita Vittone, Eric Wicklund, Steve White
- **CEPA/APS (Consulting and Design)**
Jim Kowalkowski, Marc Paterno
- **CSS/DSG ~ 3 FTE**
Richard Jetton, Steven Kovich, Anil Kumar, Svetlana Lebedeva, Nelly Stanfield,
- **CD/PPD ~ 2 FTE**
Ray Culbertson - on-line liaison
Bill Badgett, Serguei Bourov, Jeff Schmidt, Donatella Torretta

Applications

☞ Applications and Coordinators (Application is a relational database and API (C++ or Java) that allows to manipulate the contents of the database)

- Hardware Bill Badgett (PPD/CDF)
Donatella Torretta (PPD/CDF)
- RunConfiguration Bill Badgett (PPD/CDF)
Donatella Torretta (PPD/CDF)
- Trigger Tom Wright (University of Michigan)
Donatella Torretta (PPD/CDF)
- Calibrations Matt Martin (Johns Hopkins)
- Slow Control Margherita Vittone (CD)
- Data File Catalog Larry Kirsch (Brandeis University)
Dmitry Litvintsev (CD/CDF)
- SAM Rick St.Denis et al.

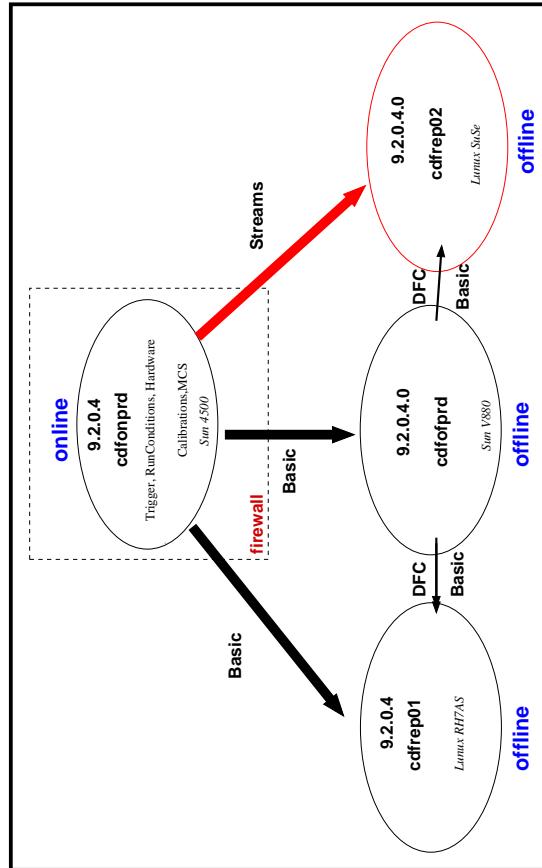




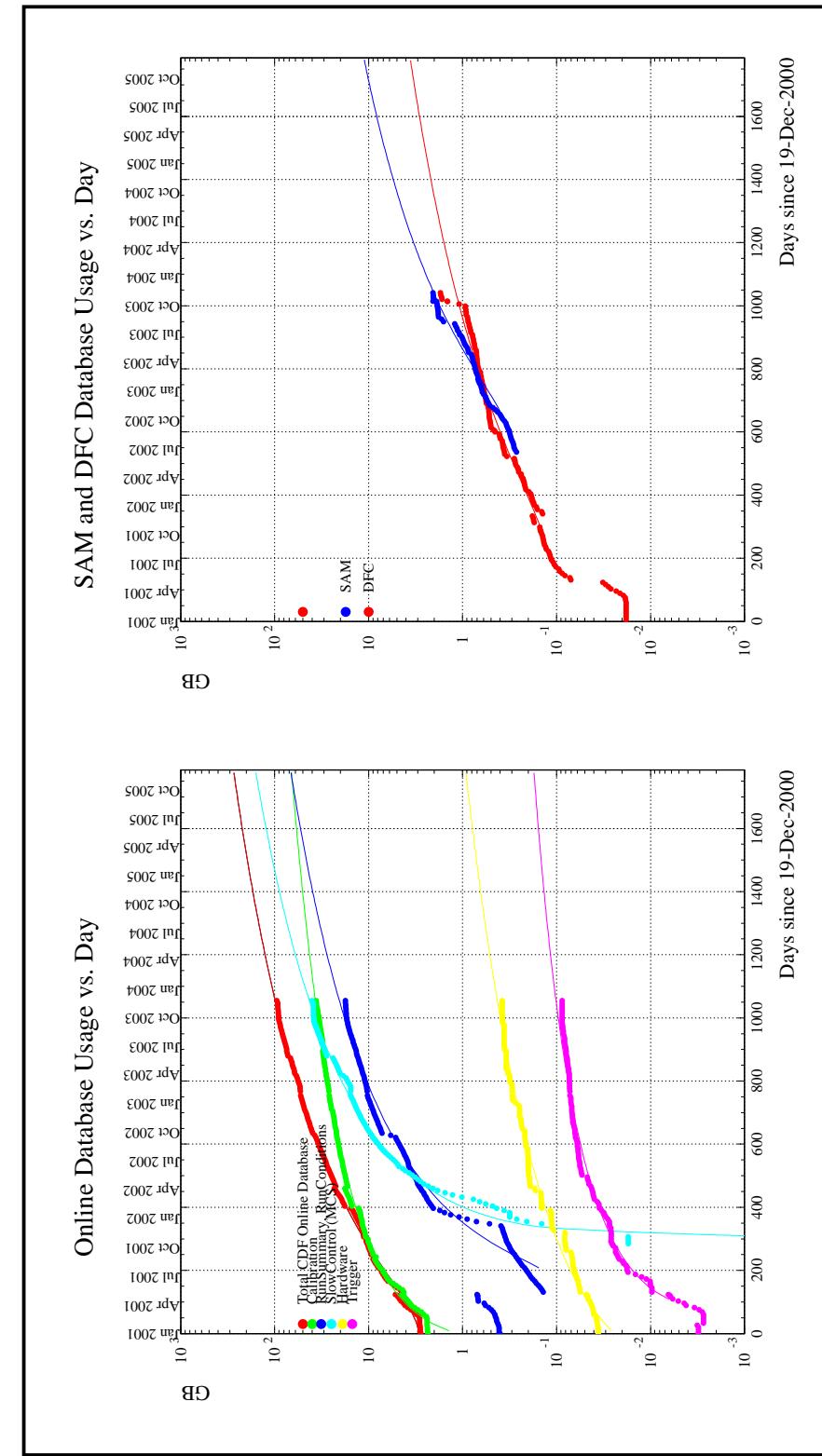
Oracle Servers

name	OS	CPU	RAM	Disk	Oracle	instance
b0dau35	Solaris 2.7	2x400 MHz USparc	1 GB	2.4 TB	9.2.0.4	cdfondprd
b0dau36	Solaris 2.9	2x400 MHz USparc	4 GB	1.6 TB	9.2.0.4	cdfonddev/int
fcdfora4	Solaris 2.9	8x900 MHz USparc	32 GB	2.0 TB	9.2.0.4	cdfofprd
fcdfora1	Solaris 2.9	4x400 MHz USparc	2.5 GB	1.0 TB	9.2.0.4	cdfofdev/int
fcdflnx1	RH AS	4x700 MHz PIII Xeon	4 GB	1 TB	9.2.0.4	cdfrepo1
fcdfora3	SUSE ES 8	2x2.8 GHz PIV Xeon	6 GB	1 TB	9.2.0.4	cdfrepo2
fcdffdata012	SUSE ES 8	? GHz PIII Xeon	?? GB	?? TB	9.2.0.4	dev/val

fcdfora2 goes to on-line to serve as fail-over in case of hardware failure on b0dau35

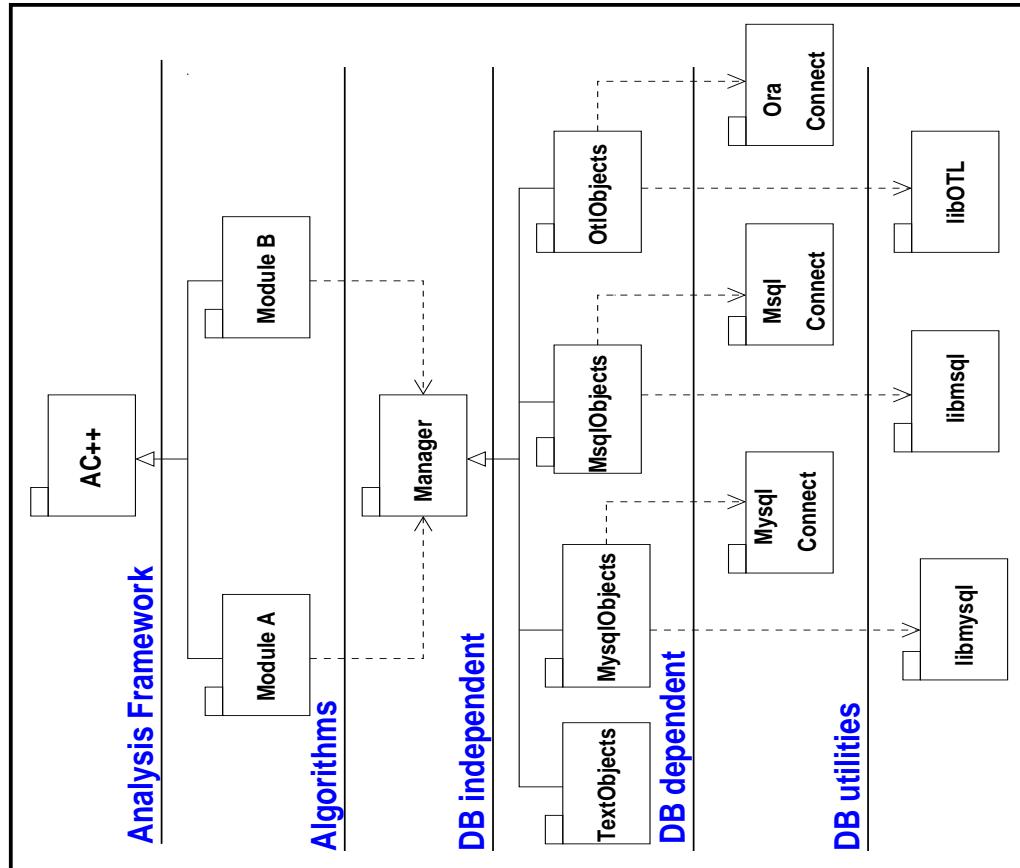


DB growth analysis



DB Access

- data stored in DBs are available to reconstruction and analysis algorithms via DB independent database management layer designed by Jim Kowalkowski (DBManager)
- DBManager provides two APIs:
 - back-end transient to persistent mapping API, *IoPackage*
 - template based front-end *Manager<OBJ,KEY>* that provides common put/get methods on transient objects.
- transient object instances can be cached by key value to configurable depth
- transient object definitions, *Mapper*, *Manager*<> and *Handle*<> classes are generated from simple Java description of persistent data using CodeGen suite.
- statically linked executables contain a lot of unused code (code bloat)
- problem of keeping *Mapper* objects in sync for all DB implementations



CDF code accesses Oracle servers directly



DB Access

- in principle we are satisfied with DB access API
- existing API allows to change underlying DB access mechanism w/o any changes to client code.
- this opens a window of opportunities
 - should CDF become interested in middle tier DB access it can be accommodated by providing corresponding back-end
 - CDF is interested in substituting multiple free-ware back-ends (mSQL or MySQL) with single interface to ODBC
 - we want to retain ORACLE layer to continue to rely on Oracle specific features like triggers



Projects

- Free-ware DBMS
 - DB Monitoring
 - DB metering
 - Code Reviews
 - Schema Reviews
 - DBManager based API development for trigger, hardware and RunConfigurations
 - Codegen Rewrite
 - Calibration Task List
 - CDF SAM
 - DB Browser
 - N-tier pilot project
 - Data Guard for on-line
- S. Lebedeva, D. Litvintsev
Y. Guo (Fermilab), J. Kowalkowski (Fermilab)
 - E. Wicklund (Fermilab)
 - D. Box (Fermilab), N. Stanfield (Fermilab)
A. Kreymer (Fermilab)
 - J. Kowalkowski (Fermilab), D. Box (Fermilab)
plus experienced CDF programmers
A. Kumar (Fermilab), N. Stanfield (Fermilab)
 - J. Trumbo (Fermilab)
 - CDF Helsinki group, overall up to 7 FTE
 - D. Box (Fermilab), J. Cranshaw (TTU)
 - D. Box (Fermilab), J. Cranshaw (TTU)
 - Y. Guo (Fermilab), J. Kowalkowski (Fermilab)
 - L. Sexton-Kennedy (Fermilab)
 - R. Herber (Fermilab), A. Kreymer (Fermilab)
 - R. Kennedy (Fermilab), D. Litvintsev (Fermilab)
 - J. Tseng (Fermilab)
 - people from Glasgow, Oxford, UCL TTU, Karsruhe University, Rutgers University
 - R. Herber (Fermilab)
 - Jim Kowalkowski, Marc Paterno
Barry Blumenfeld, Mark Mathis, Petar Maksimovic
Ray Culbertson



Issues from previous taking stock

- DB server overloads
- Dangling connections
- Poor performance

to address these issue several projects have been spawned:

- **DB Monitoring**
<http://wwwserver2.fnal.gov/cfdocs/projectsdb/projdetail.cfm?ProjectID=121>
- **Connection code review**
CDF/PUB/COMP_UPG/PUBLIC/6179
- **DB metering code**
- **DB schema review**
<http://wwwserver2.fnal.gov/cfdocs/projectsdb/projdetail.cfm?ProjectID=16>
- **Code reviews**



DB monitoring

Yuyi Guo, Jim Kowalkowski, Margherita Vittone,
Eric Wicklund, Andrew Hamilton

- Error reporting code has been added to DBManager
- Based on ErrorLogger it reports to separate logging server
- User control of detail level on per-job basis.
- Use as our primary tool for finding out out db usage patterns
- CDF will need certain functionalities added expanded from time to time like the addition of new monitored events (query durations, module names etc) Summary tables in history plots
- Need to add count of number of accesses to a table and amount of time spent accessing a table (by table name). Summary information per job has to be recorded in the logging server
- Very successfull and useful project it has become joint CD/CDF/D0 project.
Allows to catch inefficient/buggy code



Stream Replication

<http://wwwserver2.fnal.gov/cfdocs/projectsdb/projdetail.cfm?ProjectID=109>
A. Kumar, N. Stanfield

- Oracle stream replication promises to provide a solution for DB scalability issues as the number of users and the load grows
- Planned to fully implement stream replication by 1 Apr, 2003, but faced performance and stability issues. Working with Oracle on resolution
- Work on stream replication was put on hold to make room for offline production h/w replacement
- **This project will resume as soon as we get test stand server (being purchased). This is high stake project for CDF as it will allow us to create scalable, distributed DB system, suitable for GRID computing**



Freeware DBMS

Motivation

- Svetlana Lebedeva, Dmitry Litvintsev, code from Rich Hughes and Dave Waters
 - users doing analysis at remote institutions would clearly benefit from being able to read from local database
 - one of the requirements to CDF Data Base access layer was to provide back-end to freeware DB implementations. The choice at the time was mSQL
 - all CDF DB applications accessible from off-line reconstruction or analysis code support mSQL
 - there is a positive experience with running off mSQL database at Rutgers University (DFC only)
 - database export using mSQL has been worked out by Mark Lancaster but is now in limbo
 - MySQL back end is very similar to mSQL back end. It was implemented by Dave Waters.
 - MySQL is widely used database
 - advantage of using public domain products to populate MySQL from Oracle
 - advantage of having replication features built into recent versions of MySQL (not the case with mSQL)



Freeware DBMS

Delivered

- MySQL and PostgreSQL packaged in UPS/UPD
- Java code that replicates Oracle data into MySQL/PostgreSQL on demand is packaged in publicly accessible cvs repository
- CDF has central MySQL server that contains replica if offline production database (ncdf141)
- Documentation on installation and population of local MySQL server:
<http://www-css.fnal.gov/dsg/external/freeware/>
- Physicists at Karsruhe University were able to setup, populate and run against their local MySQL replica machine. The initial results are encouraging (factor of 2.5 faster)
- CDF DB group and CSS/DSG are in the process of setting up next stage of this project, that would result in having central Fermilab MySQL server which is kept up to date with Oracle. Off-site institutions will use freeware means of replicating data from that central server

Codegen

CodeGen Rewrite
People: Dennis Box (Fermilab)

- ☞ CodeGen rewrite is complete and repositored in cvs
- ☞ This is OTL and Text versions of CodeGen
- ☞ Dennis continues to work on having CodeGen to generate ODBC back-end code.
- ☞ This project nears completion
- ☞ **Next will be code generation for n-tier layer**





Development/maintenance

API development

- Improvements of CalibrationManager (Matt Martin)
- ConfigManager (Does Bill support this module?)
- **TriggerMap rewrite (Dmitry)**
- Trigger summary in the file header (Dmitry, Liz, Kevin)
- Rework of good run bits implementation (combination of Fedor, Mario, Dmitry, Bill, Donatella)
- Luminosity calculation with dynamic prescales (Dmitry)
- AC++ SAM interface (stefan ?)
- **TGRSim++ fix (Risto, Tom) Done!**
- TGRSim++ fix (Risto, Tom) Done!
- freeware back-end implementation where necessary
- DQM interface (Mario)

schema development

- re-design of tables in HDWDB (Donatella)
- re-visit design of MCS (Margherita?)
- design and implementation of good run lists for DOM (Mario?)
- SAM/DFC migration (this seems like done project)



On-line DB failover

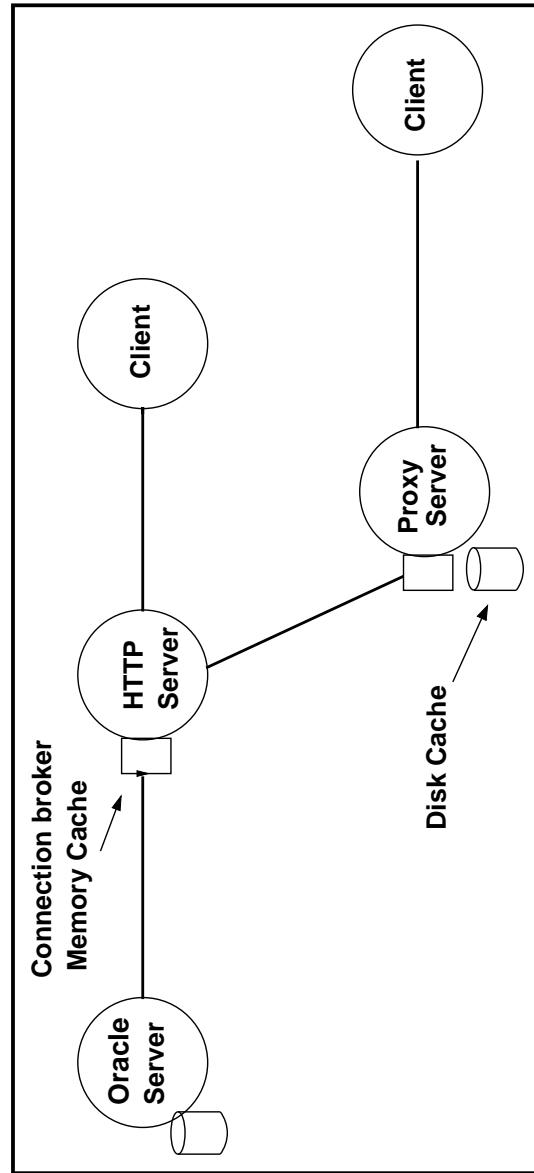
- Installation of fcfdora2 to serve as failover host for b0dau35 in case of hardware (CPU/box) failure. (Ray, DBAs, on-line SysAdmins)
- natural disasters (elementals), human errors or major Oracle failures scenarios are ought to be addressed



N-tier

Jim Kowalkowski, Marc Paterno, Dennis Box, Barry Blumenfeld, Mark Mathis,
Petar Maksimovic

- provides DB connection management
- decouples DB and client interfaces
- configuration flexibility
- secondary sourcing of the data at the remote site
- caching reduced bandwidth to Oracle server
- provides scalable distributed solution w/o Oracle licensing complications





Conclusion

- ☞ DBA services provided by Anil, Nelly and Julie are essential for:
 - keep replication working
 - setting up streams replication
 - 24x7 support of main production servers
 - table analysis
 - training new people
- ☞ CDF DB API layer does not create major operational problems or adversely affects reproducibility of physics results
- ☞ Although this API is not suitable for distributed computing. CDF DB group has adopted multipronged approach to provide scalable, distributed database system to accomodate grid environment:
 - reduction of access to DB by storing constant information on the data files (event catalog, trigger summary etc.)
 - development of Oracle stream replication.
 - installation of freeware DB servers at remote institutions populated on demand from central freeware source
 - implementation of n-tier architecture
- ☞ programming support from Dennis and design support from Jim and others is required to both:
 - bring development projects to completion
 - keep existing applications in good shape. This code is being heavily pounded upon by users and there is always a need for fixes or extended of functionality